# Grid Economics: 10 Lessons from Finance

Giorgos Cheliotis<sup>†</sup>, Chris Kenyon<sup>†</sup> and Rajkumar Buyya <sup>‡</sup>

†IBM Research Zürich Research Lab Säumerstrasse 4 8803 Rüschlikon, Switzerland {gic|chk}@zurich.ibm.com

‡Grid Computing and Distributed Systems (GRIDS) Lab Dept. of Computer Science and Software Engineering University of Melbourne 221 Bouverie St. Melbourne, Australia raj@cs.mu.oz.au

April 24, 2003

### Abstract

Today Grid technology is finding its way out of the academic incubator and into commercial environments. The interest in on-demand utility computing is larger then ever before in the industry and Grid tools are seen as an important enabler of this business model. The process of Grid commercialization inevitably brings financial issues to the fore. When multiple budget entities share resources, allocation decisions in resource management become also financial decisions. For the potentially large and complex systems enabled by Grids, equivalently robust financial engineering is needed. We draw parallels to the financial and commodity markets and outline 10 lessons learnt in the long history of asset management and decision-support for these markets. We show the relevance of these lessons for Grid commercialization and suggest specific issues that need to be resolved before (virtual) organizations can realize the theoretical value of Grid computing.

**Keywords:** Grid computing, commercialization, commodities, resource trading, virtual organization

### 1 Introduction

Grid technology enables sharing of computing resources within and between organizations. As a result of this, Grid resources will be (re-)allocated dynamically in response to changing needs. This (re-)allocation will very often be across budget boundaries, even within single organizations, thus bringing financial questions to the fore. What is a Grid computing resource worth today? What will it be worth tomorrow?

In this article we illustrate the organizational and system design challenges in the economic and financial domains that are (and will be) surfacing from Grid technology adoption. We use insight from economics and finance to draw lessons for system designers and adopters to address these new challenges.

This article is based on the content of a half-day tutorial given at GGF5 in Edinburgh on the same topic and the feedback that we received.

# 2 Sharing

Grids are conceived as distributed cross organizational systems where users share resources at a scale and with an ease that was not possible before. First Grid prototypes have been successful in enabling this scale of sharing academically but the business environment with its time and financial pressures emphasizes the question: "now you have a Grid, who gets what, when and for how much?".

#### 2.1 The Tragedy of the Commons

Usage increases whenever there is an increased need for computation, i.e. when the marginal benefit to the user of utilizing an extra unit of computation exceeds the cost of consuming that unit. In an uncontrolled or free-for-all situation this marginal cost is very low.

The above statement seems to imply that only individual users' budgets and preferences are relevant in determining resource value and this is generally true in the absence of what economists call (network) externalities. An externality can be defined as the impact of one person's actions on the well-being of a bystander and can be positive or negative. Examples of negative externalities are:

- Overgrazing of land held in common
- Adding many repeat keyword entries in HTML pages so as to artificially increase ranking on early search engines



Figure 1: The Tragedy<sup>1</sup> of the Commons.

This socioeconomic phenomenon whereby the individually "rational" actions of members of a population have a negative impact on the *entire* population is often called "the tragedy of the commons" (see Figure 1).

Common recipes for dealing with this issue target the internalization of negative externalities into every individual's decision process. This can be achieved by taxation, regulation (e.g. TCP congestion control), private solutions or prices for access rights, e.g. permits.

Shared Grid infrastructure is particularly prone to negative externalities because there is currently no scalable and dynamic standard mechanism for limiting system (ab)use. Local priority rules are efficient in returning Grids to their pre-Grid, i.e. non-shared, state whilst the shared spaces suffer from the Tragedy of the Commons. Static policies are particularly inappropriate for dynamic Virtual Organizations and do not scale well as the number of participating entities increase. Pricing access to Grid resources and permitting resale is a direct and scalable way to preclude such a tragedy of the commons for Grid deployments that deserves serious consideration.

#### 2.2 Resource value is dynamic and must be communicated to consumers

Given even the most cursory awareness of conventional resources and commodities such as copper, electricity and petrol (gas) it is clear that resource value at the wholesale level is dynamic. What is perhaps less clear to some casual observers is that resources on Grids have dynamic value.

Value derives from a combination of need and scarcity. User needs are not constant, they change over time and the changes also depend on the timescale and granularity of observation. During a project lifecycle a single user working on that project will have varying workloads in different phases of development. The number of projects that a user is involved in also changes with time. Needs are also driven by external and irregular events, e.g. reaction to advertising campaigns, seasonality, request-for-bids that require data analysis.

Variations in user needs change resource value very little if the resources are not scarce, i.e. if the capacity of the shared infrastructure is never exhausted. However this happy state is rarely present for users with computationally heavy applications.

Financial and commodity markets have long established methods to communicate and prioritize competing needs of many individuals: prices. These are understandable to every single participant and allow the participants to take effective action.

Isolating users from price dynamics makes sense when they never see, or cause, scarcity, i.e. when they have low and uncorrelated needs. For example bread price dynamics at supermarkets have little relation to corn futures markets. On the other hand

 $<sup>^{1}</sup>A G(r)eek Tragedy \dots$ 

electricity companies seek methods to pass intra-day price dynamics on to consumers because of the enormous loads consumers produce through correlated responses to events (e.g. extremes of temperature) even though each individual consumes little relative to the capacity of an electricity generator. Most users of Grid infrastructures are heavy resource consumers almost by definition so dynamic prices will be used at some level.

#### 2.3 Price formation mechanisms are easy to implement but difficult to design

The mapping of needs to prices, "price formation", has no single solution but there is an extensive body of work precisely on this topic: auctions [Kle99]. Whilst prices must be fed back to users, there is no corresponding need for the price formation mechanism to be visible to users. This can be handled for the most part by automated software: but a mechanism is still required and there are significant design challenges for it.

The lesson from auction theory and practice is that the choice of price formation mechanism can either promote market efficiency or hamper it. Generally it is difficult to achieve a balance between the needs of producers and consumers. Recent examples that illustrate this difficulty very well are the UMTS (Universal Mobile Telecommunications System, i.e. 3G mobile telephony spectrum) auctions [Kle, Wol].

High profile auctions for 3G licenses have been carried out in many European countries. Two distinct problems arose in these auctions: bidder busts ("winner's curse") and auctioneer flops. 3G auctions in Germany and the UK yielded enormous profits for the local authorities at the expense of the bidders, whereas in Switzerland, the Netherlands, Italy and Austria prices remained well below expectations, disappointing the respective auctioneers.

In Grids we want to avoid the winner's curse, also resources are perishable (capacity not used now is worthless in the next moment), needs are dynamic and applications require bundles with multiple units of items (CPU, RAM, permanent storage, network bandwidth). Potentially suitable auction models for Grid resources include continuous double auctions, Vickrey, Dutch, multi-unit and multi-item (or combinatorial) auctions. However, individually these approaches do not offer a comprehensive and precise price formation solution. In any case the optimality of an auction mechanism will always depend on the particular deployment environment, there are no one-size-fits-all solutions.

# 2.4 Real money is better than funny money

An issue which concerns the Grid community today is the definition of a Grid currency. The issue is more important for Grids than for earlier distributed systems because commercial Grids cross budget boundaries. In addition managers will face the issue of whether to buy resources on accessible Grids or boxes, and also whether to make their boxes available to the Grids to which their organization is linked.

Grids and resources are generally heterogeneous and potentially of arbitrary scale. Scale and heterogeneity are exactly the drivers which led to the establishment of standard monetary units and currency exchange rates in the real economy.

The administration of a particular Grid may choose to introduce prices for a local artificial currency. The administration must then act as a national bank guaranteeing the convertibility of the currency into units of value, i.e. resources or real money. Now who sets the exchange rates and to which unit of value? A currency board? A fixed exchange rate?

Most IT administrations will quickly choose to jump the intermediate step of an artificial currency with its trust and convertibility problems and use real money straight away. Using a real currency for Grid resources additional brings the following benefits: buy/build/lease or upgrade/retire decisions are simplified and the allocation of IT budgets is directly meaningful.

## **3** Resource Allocation

We now try to answer the question: what sort of allocations should be made from an economic perspective?

# 3.1 Value depends on property rights (QoS)

What QoS is required for tradeable value? Most IT systems today do not support hard QoS guarantees, that is they do not guarantee the properties of a service which influence user experience. Often best-effort service is provided. Approaches which go beyond best-effort typically introduce job/packet marking so that different priorities can be assigned to different tasks [BEP+96, FH98]. How much *better* the service will be for differentiated service classes is generally hard to determine in advance for large-scale heterogeneous systems and even harder to characterize in absolute terms.

Despite the difficulties of guaranteeing QoS (especially end-to-end), Grid commercialization requires



Figure 2: Best-Effort.

guaranteed property rights at the level at which pricing is done. In a commercial environment, buyers can expect sellers to optimize what they deliver against the QoS guarantees that they provide. We should also make clear at this point that best-effort service has near-zero economic value. In fact the value would be exactly zero if it were not for the assumption that there is a common understanding between the buyer and seller of the service on the quality level to be delivered (see Figure 2).

Advocates of grid computing envision dynamic nearreal-time negotiation and provisioning of distributed resources. Existing financial and commodity markets which operate at electronic speed rely on the use of extremely detailed contracts. Complexity is no barrier to value for a good. The definitions of some resources traded on the Chicago Mercantile Exchange (CME) run for many pages and even then reference external tests and standards.

Computers and applications may be complex but they also have unambiguous definitions. This level of detail is necessary to create the appropriate confidence among users of a highly distributed crossorganizational system that what they get is exactly what they expected to receive. In some cases a tradable asset must be described in statistical terms. This has been applied to cycle-scavenging [KC02].

# 3.2 Futures markets dominate when assets are not storable

We mentioned earlier that grid resources are not storable, in the sense that capacity not used today cannot be put aside for future use. The most significant non-IT resource which is also non-storable is electrical power (with the notable exceptions of hydroelectric and pumped storage). In electricity markets, as in several others for non-storables, contracts for future delivery (forward or futures contracts) are the most used and have much higher trading volumes than those for immediate delivery (spot contracts). The explanation for this is that participants want to manage the risk of price/availability uncertainty by fixing the price and availability of a resource in advance. See [Hul00] for an introduction to futures markets.

Practical Grid markets will revolve around reservations (forwards) not spot markets. The experience of electricity markets is clear (e.g. California, UK) and has led to their redesign with the aim to move everything possible off the spot market and onto the reservation markets. Without inventories volatility has no real upper limit for non-storable resources. High volatility is not a desirable characteristic for most resource buyers or sellers.



Figure 3: No Reservations?

The fact that work has started within the GGF for supporting advance reservations is encouraging [RS02] (see also Figure 3).

# 4 Markets and Processes

What can we learn from the design of established markets that will help in building efficient and liquid Grid markets? What processes are required for effective exploitation of such large-scale and dynamic situations for business advantage?

#### 4.1 Practical incentives are a precondition for resource sharing and exchange

In the period 1999-2001 over 1000 new internet based B2B exchanges were set up (according to IDC). Almost all failed. These did not fail because markets have poor theoretical properties, but because their specific value proposition was unconvincing. This was made manifest in "low liquidity", i.e. no activity.

The success stories in B2B markets are mostly in procurement, for very large companies, and specialized National or regional exchanges for electricity. These have one thing in common: force not persuasion to get people to sign up. Some of the results may be good for everyone, but the startup adoption costs must still be paid. The other area of outstanding B2B exchange success is the traditional financial and commodity markets with their vast turnover.

Where does Grid sharing and exchange fit into this spectrum of experience? It is outside the scope of this paper to answer in detail but certainly the value proposition of cost savings and greater flexibility is generally accepted. Commoditization is also accepted: there are many fewer computer flavors than there are different companies on, say, the New York Stock Exchange.

A company can decide to migrate to an internal market in the same way that it can decide to outsource. This is an executive decision to be made on business grounds: units may protest for whatever reason but the needs of the business are the deciding factor.

Public Grid exchanges are unlikely in the short to medium term because of complexity of implementation and of the required changes in business processes. Within a single company, or a closed group, the prospects for having appropriate incentives to overcome the startup costs and general inertia are much higher.

#### 4.2 Trust builds on mutual interest

What is a good trust model for Grid computing? We note that trust is different from security and we are concerned here with trust. Security is just one enabler of trust (see Figure 4).

A good trust model for an online bookstore is not the same as a good trust model for a financial exchange. In fact online bookstores effectively outsource their trust model to credit card companies for the most part. All the bookstore has to do is provide a basic level of security.



Figure 4: Trusted or just... secure?

A financial exchange such as the CME has a more complex trust model. Firstly, all transactions on the exchange between different parties are guaranteed by the exchange not by the individual parties. Thus the traders only need to trust the exchange to do business, not each other. This improves liquidity enormously by simplifying the trader's trust model. On the other hand the exchange trusts the traders because it monitors their actions and requires them to provide a (cash-equivalent) deposit which is a function of the risk that each trader represents to the exchange for default. That is, the exchange trusts the traders because it has their money. All other people wishing to trade must do so via the traders. Thus we see a two-tier trust model with distributed risk.

Systems without proportional consequences do not engender trust: this is why contracts exist and are specific and clear methods to invoke financial and legal penalties for improper actions. Grids require trust models with proportional consequences, adapted to their environments (e.g. single company, group, etc.).

#### 4.3 Large-scale dynamic systems require appropriate process tools

Do financial firms optimize their allocations (portfolios) by hand? After they have used their optimization tools, perhaps, but certainly not beforehand. A typical portfolio directed by a fund manager can easily run to hundreds of stocks selected according to maximizing a specific objective and limited by equally precise constraints on number of stocks to hold, position limits, cashflow obligations that must be met on specific dates, hedging against worst case scenarios, etc.

The dynamic system enabled and embodied by Grid



Figure 5: User in need of process tools.

computing for a typical large company is a significant challenge for users to be able to exploit efficiently and economically. Without sophisticated tools users will find that the more potentially useful the Grid system is the less practically usable it will be (see Figure 5).

The process tools in the Grid space must enable users to express their resource needs in a simple way together with the users' uncertainties about these needs over time. They should also enable resource trading (buying and selling of blocks of time and capacity) and capture the effective price dynamics of both spot and futures prices together with changing availabilities. Building such tools which integrate budgets and business objectives with resource allocation decisions may seem overly ambitious but it is a situation that is tackled every day for fund managers balancing opportunities and obligations over time (see [Nef00, Mar59, BL97]).

#### 4.4 Economic engineering is required to realize potential value

When making a business case for Grid technology adoption the following arguments are common: increased utilization, cost savings, greater allocation flexibility, feasibility of previously impossibly computational tasks, etc.

These may be theoretically possible but to what ex-

tent can an organization practically realize these potential values of a Grid deployment?

As mentioned in the previous section on futures markets a market and resource product structures are engineered to achieve results: *laissez-faire* alone is not enough. This is even truer in cases where markets will be created for intra-Grids, that is for the sole purpose of achieving an efficient resource allocation internally.

One set of economic engineering questions, that need to be answered in Grid deployments to realize theoretical Grid value, center on market engineering. Questions include the following, which are just a small selection. Should the IT department of an organization be operated as a profit or cost center? How much reselling of resources should be allowed? Should shortselling be allowed? Is speculation permitted? What level of financial and project risk are users and departments permitted? Are bilateral trades permitted?

Engineering of (resource) products is another area requiring design. Spot and forward contracts (reservations) may be useful for describing and controlling the theoretical basis of value. These can be automatically assembled to matching user, application, and department requirement profiles both using process tools (multi-stage stochastic portfolio optimization). A complementary approach for providers is to manually design resource product packages incorporating spots, forwards, and quantity and timing flexibility to match user, application and department needs, i.e. construct derivative products.

#### 5 Grid Practice

We will now briefly comment on a couple of computational environments in use today that start to implement significant aspects of the lessons above.

#### 5.1 Nimrod-G

Nimrod-G is a resource management and scheduling system that supports deadline and budget-constrained algorithms [BGA01]. It has been used for scheduling parameter sweep applications on global Grids. Application experts can create a plan for parameter studies and use the Nimrod-G broker to handle all the issues related to the job management and execution, including resource discovery, mapping jobs to appropriate resources, data and code staging and gathering results from multiple Grid nodes back to the home node. Depending on the user's requirements, it dynamically leases Grid services at runtime based on their availability, capability, and cost.

The inclusion of budget and deadline constraints in an easy-to-use GUI is a first step towards the creation of decision-support process tools for Grid users. Nimrod-G is a significant proof point for economicsrelated and decision-support process tool issues in the Grid community. The newly started Gridbus project at the University of Melbourne, building on this work, is aiming to include several new features such as endto-end QoS.

#### 5.2 ZetaGrid

ZetaGrid is an open source and platform independent secure cycle-scavenging system developed by the IBM Development Laboratory in Böblingen. It provides a secure Java kernel, which does not allow tampering by the host system, and secures its communications and activities by restricted layer access with digital signatures and key establishment protocols.

ZetaGrid serves as an example of a cycle-scavenging system which on the one hand protects the participating machines from malicious jobs/users and on the other hand protects these jobs and their data from eavesdropping and modification. It has thus made the first steps towards creating a trusted environment for large-scale grid computing.

# 6 Conclusion

The next wave of Grid computing challenges will be dominated by business-enablement issues rather than those of technological enablement. We have provided here a set of lessons from the viewpoint of financial engineering with which to meet these emerging challenges. We thus hope to contribute in the shaping of a new research agenda for Grid computing.

### 7 URL

- 1. California electricity market re-design, http://www.caiso.com
- 2. The Chicago Mercantile Exchange, http://www.cme.com
- 3. The CME Rulebook, http://www.cmerulebook.com
- 4. The GridBus Project, http://www.gridbus.org
- 5. IBM Zurich Grid Economics, http://www.zurich.ibm.com/grideconomics

#### this work, chasend [BGA01]

References

[BGA01] R. Buyya, J. Giddy, and D. Abramson. A Case for Economy Grid Architecture for Service-Oriented Grid Computing. 10th IEEE International Heterogeneous Computing Workshop, April 2001. San Francisco, California, USA.

[BEP<sup>+</sup>96] J. Blazewicz, K. Ecker, E. Pesch, G. Schmidt,

and J. Weglarz. Scheduling Computer and Man-

ufacturing Processes. Springer-Verlag, 1996.

- [BL97] J. Birge and F. Louveaux. Introduction to Stochastic Programming. Springer, New York, NY, 1997.
- [FH98] P. Ferguson and G. Huston. Quality of Service on the Internet: Fact, Fiction or Compromise? In *Inet 98*, 1998.
- [Hul00] J. Hull. Options, Futures, and Other Derivatives, Fourth Edition. Prentice Hall, 2000.
- [KC02] C. Kenyon and G. Cheliotis. Creating Services with Hard Guarantees from Cycle-Harvesting Systems. IBM Research Report RZ 3461, available at http://www.research.ibm.com. Submitted to CCGrid 2003, Tokyo, May 2003., 2002.
- [Kle] P. Klemperer. How (Not) to Run Auctions: The European 3G Telecom Auctions. Available at http://www.paulklemperer.org.
- [Kle99] P. Klemperer. Auction Theory: A Guide to the Literature. Journal of Economic Surveys, Vol. 13(3), pages 227–286, July 1999.
- [Mar59] H. Markowitz. Portfolio Selection: Efficient Diversification of Investments. John Wiley and Sons, NY, 1959.
- [Nef00] S. Neftci. An Introduction to the Mathematics of Financial Derivatives, 2nd Edition. Academic Press, 2000.
- [RS02] A. Roy and V. Sander. Advance Reservation API. GGF draft, available online: http://www.gridforum.org, 2002.
- [Wol] E. Wolfstetter. The Swiss UMTS Spectrum Auction Flop: Bad Luck or Bad Design? Available online: www.wiwi.huberlin.de/wt1/lectures/design/0102/swissau.pdf.

6. ZetaGrid, http://www.zetagrid.net